

APLIKASI REKOMENDASI TAG PADA SITUS BERBAGI GAMBAR FLICKR®

Intan Yuniar Purbasari¹, Hongyun Cai², Angeli Diane Lao², Iman Mohamed Al-Rawahi²

¹Jurusan Teknik Informatika, Fakultas Teknologi Industri, UPN "Veteran" Jatim

²School of ITEE, The University of Queensland, Australia

Jl. Rungkut Madya, Surabaya

email : intan@upnvjatim.ac.id

Abstrak. Dalam sebuah situs berbagi gambar, ketika pengguna akan mengunggah sebuah gambar, biasanya akan memberikan tag/keterangan pada gambar tersebut. Tujuannya agar gambar dapat ditemukan kembali oleh pemiliknya atau oleh pengguna lain yang memiliki minat yang sama terhadap topik gambar tersebut. Aplikasi ini menerapkan teknik *Association Rule Mining* pada tag sebuah situs berbagi gambar untuk memberikan rekomendasi pada pengunggah gambar tentang tag lain yang dapat ditambahkan pada gambar yang akan diunggah berdasarkan pada pasangan tag yang sering muncul pada database gambar yang ada, dengan mengasumsikan tag yang digunakan adalah dalam bahasa Inggris. Dalam percobaan berskala kecil ini, diambil tag awal "animal" sebagai topik utama pencarian untuk kemudian dicari tag lain yang berasosiasi dengan tag tersebut. Aplikasi ini dapat menghasilkan rekomendasi tag lain ketika pengguna melakukan pencarian sebuah tag yang bertemakan "animal".

Kata kunci: rekomendasi tag, association rule mining, flickr

Tag adalah keterangan tertulis yang ditambahkan kepada sebuah gambar untuk memberikan informasi tentang isi gambar tersebut. Tag pada gambar berfungsi sama seperti kata kunci pada sebuah dokumen. Ketika pemilik gambar mengunggah gambar miliknya ke sebuah situs berbagi gambar, biasanya ia akan memberikan tag pada gambar tersebut. Tujuannya selain untuk memudahkan pemilik itu sendiri untuk menemukan gambar tersebut, juga agar pengguna lain dari situs berbagi gambar yang memiliki minat yang sama dapat mencari gambar yang relevan dengan minatnya[1]. Pemberian tag pada gambar juga akan memudahkan *search engine* untuk merambah gambar sesuai dengan query yang dimasukkan oleh pengguna.

Pada kenyataannya, menurut [2], setidaknya 20% dari gambar yang diunggah secara publik di situs berbagi gambar Flickr® tidak memiliki tag sama sekali dan 64% dari gambar yang memiliki tag hanya terdiri dari 1-3 tag saja [3]. Hal ini dapat terjadi dikarenakan pemilik gambar enggan memberikan tag yang bermanfaat pada gambar yang diunggahnya karena dianggap menyita waktu tambahan

untuk memikirkan tag apa saja yang tepat untuk mendeskripsikan gambar tersebut.

Penambahan fitur rekomendasi tag telah diteliti sebelumnya dalam [2] dengan menggunakan algoritma klasifikasi untuk melatih dan mempelajari data tag dan melakukan prediksi untuk tag yang baru. Dalam [4] juga telah dibuat sebuah sistem yang dapat memprediksi tag dalam Flickr® yang mengkombinasikan fitur linguistik dan juga vision. Hasil yang didapat menunjukkan bahwa algoritma yang digunakan menunjukkan performa yang tinggi dalam merekomendasikan tag kepada pengguna.

Penelitian ini bertujuan untuk menghasilkan rekomendasi tag pada gambar yang diunggah di situs berbagi gambar Flickr®. Metode yang digunakan adalah teknik *Association Rule Mining* yakni algoritma *Apriori* untuk menemukan pola pemberian tag oleh pengguna dan *K-Means clustering* untuk mengelompokkan gambar sebelum dilakukan proses *rule mining*. Dataset yang digunakan diambil dari situs Flickr® dengan melakukan *crawling* terhadap tag dari 8000 gambar.

1. METODOLOGI

Penelitian ini menggunakan algoritma Apriori sebagai algoritma untuk melakukan *association rule mining* dan algoritma K-Means untuk mengelompokkan data sebelum dilakukan proses *rule mining*. Untuk melakukan *crawling* terhadap tag Flickr®, digunakan Application Programming Interface (API) untuk Flickr® yang berbasis Java, yakni FlickrJ.

1.1. Algoritma Apriori

Algoritma Apriori [5] merupakan algoritma klasik yang mencari item set yang sering muncul bersamaan serta pencarian *rule* asosiasi dari basisdata transaksional. Prosesnya dimulai dengan mengidentifikasi kemunculan item individu dalam basisdata dan secara bertahap membentuk gabungan beberapa item, yang disebut itemset, jika item-item tersebut sering muncul bersamaan.

1.2. Algoritma K-Means

Algoritma ini mengelompokkan data dengan membagi berdasarkan nilai yang terdekat dengan nilai rata-rata setiap kluster. Diberikan sejumlah data (x_1, x_2, \dots, x_n), algoritma K-Means bertujuan untuk membagi n data tersebut ke dalam k kluster (dimana $k < n$) dengan meminimalkan *sum of square* dari masing-masing kluster yang persamaannya seperti pada persamaan (1) [6]:

$$\underset{s}{\operatorname{argmin}} \sum_{i=1}^k \sum_{x_j \in S_i} \|x_j - \mu_i\|^2 \quad (1)$$

dimana μ_i adalah rata-rata dari data pada kluster S_i .

1.3. Singular Value Decomposition (SVD) dan JAMA

Dalam dunia aljabar linier, SVD adalah teknik faktorisasi matriks yang kompleks. Pada penelitian ini, SVD digunakan untuk mereduksi ordo pada sebuah pemodelan (dalam hal ini adalah matriks mapping ID dan tag). Tujuan pereduksian ini adalah untuk mengurangi derajat kebebasan dalam sebuah sistem kompleks yang akan dimodelkan sehingga akan mengurangi waktu komputasi.

Misal M adalah matriks berukuran $m \times n$ yang nilainya berasal dari kolom K . Maka bentuk formal dari faktorisasinya adalah seperti pada persamaan (2) [7]:

$$M = U \Sigma V^* \quad (2)$$

dimana U adalah matriks uniter terhadap K , matriks Σ adalah matriks diagonal $m \times n$ dengan nilai non-negatif pada diagonalnya, dan V^* adalah matriks uniter $n \times n$ yang menyatakan transpose konjugasi dari matriks uniter V yang berukuran $n \times n$.

JAMA (Java Matrix) adalah sebuah pustaka aljabar linier untuk bahasa pemrograman Java [8]. JAMA menyediakan berbagai class dan fungsi untuk operasi membuat dan memanipulasi matrix berukuran besar dan padat. Tabel 1 memuat ringkasan dari operasi pada JAMA:

Tabel 1. Ringkasan Operasi yang dapat dilakukan pada JAMA

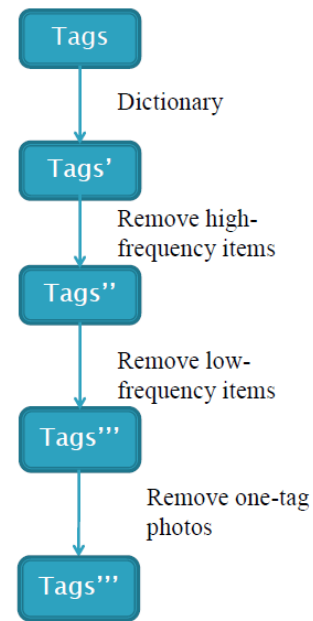
| | |
|--------------------------|--|
| Manipulasi Objek | Konstruktor Set elemen Get elemen Copy Clone |
| Operasi Dasar | Penjumlahan Pengurangan Perkalian Perkalian scalar Perkalian terhadap elemen Pembagian terhadap elemen Minus unary Transpose Bentuk normal |
| Dekomposisi | Cholesky LU QR SVD Eigenvalue simetris Eigenvalue non-simetris |
| Solusi Persamaan | Sistem non-singular Kuadrat terkecil |
| Kuantitas turunan | Bilangan kondisional Determinan Ranking Invers Pseudoinvers |

1.4. Flickr® API untuk Java – FlickrJ®

Flickr® adalah sebuah situs berbagi gambar yang cukup populer yang dikelola oleh Yahoo!®. Flickr memiliki Application Programming Interface (API) terbuka yang dapat digunakan oleh developer software untuk mengintegrasikan fungsionalitas Flickr ke dalam aplikasi apapun¹. Flickr API tersedia dalam berbagai macam bahasa pemrograman, seperti Java, ActionScript, Ruby, C, .NET, Delphi, Perl, Cold Fusion, dan lain-lain. Flickr API memungkinkan developer menggunakan class dan fungsi pada API tersebut untuk mengunggah, mengedit, dan mengunduh gambar, mengambil informasi dari gambar, mencari dan menambah topik dan juga tag gambar, dan lain-lain. FlickrJ² adalah interface API untuk bahasa pemrograman Java yang digunakan di dalam penelitian ini.

Adapun metodologi untuk membuat rekomendasi tag pada Flickr® adalah sebagai berikut:

1. Pengambilan tag dari Flickr®. Proses ini menggunakan API dari Flickr® dengan menggunakan implementasi Java sebagai bahasa pemrogramannya. Untuk menyederhanakan permasalahan, diambil satu tag sebagai awalnya (yakni tag ANIMAL) dan dilakukan *crawling* pada semua gambar di Flickr yang memiliki tag tersebut. Untuk setiap gambar yang ditemukan, diambil seluruh tag-nya dan dilakukan *crawling* kembali untuk mengambil tag dari gambar lain yang mengandung tag tersebut. Proses dilakukan secara berantai untuk mendapatkan 8000 metadata gambar. Sejumlah gambar yang tidak memiliki tag tidak disimpan dalam database, menghasilkan 6579 metadata gambar dengan 3285 tag yang tersimpan dalam database.
2. Langkah pra-proses ditunjukkan pada gambar 1.



Gambar 1. Pra-proses Tag

Kegiatan pra-proses pada tag, meliputi:

- a. Pengecekan dengan kamus untuk menyaring kata yang berbahasa Inggris saja. Cara ini dilakukan dengan membandingkan seluruh tag yang terkumpul dengan daftar kata (dalam bahasa Inggris) di dalam sebuah kamus Wiktionary[8].
- b. Pembuangan tag yang memiliki frekuensi tertinggi dan terendah agar tidak “mengotori” data. Tag yang selalu muncul di setiap gambar bukan merupakan tag yang baik untuk dilakukan proses mining, hal ini serupa dengan kemunculan kata “dan”, “ke”, “dari”, dan lain-lain pada sebuah dokumen. Oleh karena itu, tag yang memiliki frekuensi kemunculan tertinggi perlu dihilangkan. Begitu juga untuk tag hanya muncul di 1 atau 2 gambar saja (dengan kata lain memiliki frekuensi terendah). Tag dengan frekuensi tertinggi adalah “ANIMAL” dan “ANIMALS” (masing-masing muncul di 3098 dan 2935 gambar) sedangkan tag dengan frekuensi terendah adalah “CIDER” dan “CLEATS” (masing-masing muncul di 1 dan 2 gambar). Keempat tag tersebut dihapus dari database.

3. Pembuatan matriks untuk memetakan tag dengan gambar. Tag disimpan di dalam database beserta atribut “photo-id” yang mewakili ID dari gambar di Flickr. Matriks digunakan untuk memetakan gambar dengan tag, dimana baris menyimpan “photo-id” dan kolom menyimpan tag yang dimiliki oleh gambar tersebut. Jika sebuah photo-id memiliki sejumlah tag, maka di matriks akan memberikan nilai 1 pada kolom tag yang bersesuaian dan nilai 0 jika gambar tidak memiliki tag tersebut. Matriks tersebut memiliki 6579 baris dan 3281 kolom dan merupakan matriks 1-0 yang *sparse*.
4. Proses clustering untuk memudahkan proses mining tag. Karena terdapat jumlah data yang sangat besar, maka data perlu dikelompokkan (clustered) agar proses pencarian *association rule* dapat berjalan secara efisien. Gambar yang mirip (memiliki banyak kesamaan tag) dikelompokkan menjadi beberapa kluster. Algoritma yang digunakan untuk mengelompokkan gambar adalah algoritma K-Means dengan menggunakan *Euclidean Distance* sebagai pengukur jaraknya. Karena data berbentuk matriks yang *sparse*, digunakan proses *Singular Value Decomposition (SVD)* untuk memampatkan matriks dan meningkatkan efisiensi pengelompokan. Karena keterbatasan memori komputer, hanya 1000 gambar yang dapat diproses dalam satu waktu, sehingga proses pengelompokan dilakukan sebanyak 6 kali untuk memproses seluruh 6579 data. Setiap 1000 gambar dibagi dalam 4 kluster (total ada 24 kluster) dan setiap kluster terdiri dari kurang lebih 250 gambar.
5. Pencarian *association rule* pada setiap cluster untuk menghasilkan kombinasi tag yang sering muncul bersamaan. Algoritma Apriori mencari pasangan tag yang paling sering muncul bersamaan dengan menetapkan batasan *confidence* dan *support* minimum.
6. Perekomendasi tag. Tag yang direkomendasikan didasarkan pada hasil *association rule* yang ditemukan pada langkah sebelumnya. Hanya rule yang kuat yang akan dipilih untuk menghasilkan rekomendasi tag dari query tag yang dimasukkan oleh pengguna.
7. Pengambilan kembali gambar yang memiliki tag yang sama atau yang direkomendasikan. Selain menghasilkan output berupa tag hasil rekomendasi, sistem juga akan menampilkan sejumlah gambar dari Flickr yang memiliki tag yang direkomendasikan tersebut. Ini juga dilakukan melalui Flickr API dengan menggunakan method `SEARCH` pada interface `PhotoInterface`.

II. HASIL DAN PEMBAHASAN

Setelah melalui proses *mining*, beberapa pasangan tag atau rule yang berhasil ditemukan adalah:

- [cages] → [cats]
- [goat] → [homestead]
- [gastropods] → [periwinkles, sanctuary]
- [guy] → [bed]
- [cat] → [wildlife]
- [bird] → [cat, date, ontop]
- [lizards] → [cages, pirates]

Dapat dilihat pada tabel 2, bahwa di kluster 1 ditemukan jumlah *association rule* terbanyak, yakni sebanyak 165 rule. Sedangkan kluster 3 memiliki jumlah rule yang paling sedikit, yaitu 6 rule.

Dari sejumlah pasangan tag yang ditemukan, diketahui beberapa pasangan tag yang masih masuk akal, seperti “cages” dengan “cats”, “cats” dengan “wildlife”, “goat” dengan “homestead”, dan “gastropods” dengan “periwinkles”. Pasangan lain yang cukup menarik adalah “cats” dengan “birds”. Sedangkan pasangan tag “guy” dengan “bed” dianggap cukup mengejutkan untuk muncul di dalam penarikan asosiasi.

Tabel 2. Jumlah *Association Rule* yang ditemukan pada setiap kluster

| Cluster | Association Rules |
|------------------------|-------------------|
| Cluster(1): 230 photos | 165 |
| Cluster(2): 314 photos | 10 |
| Cluster(3): 229 photos | 6 |
| Cluster(4): 228 photos | 35 |

Karena gambar telah dikelompokkan menjadi beberapa kluster, maka rule yang ditemukan pada tiap cluster pun mengandung kata-kata yang sama. Sebagai contoh, rule tentang *lizards*, *cages*, dan *pirates* ditemukan di kluster 1 dan terdapat 165 variasi rule yang mengandung ketiga kata tersebut. Sedangkan rule tentang *gastropods*, *periwinkles*, dan *sanctuary* ditemukan di kluster 2 dan memiliki 10 variasi rule.

Permasalahan yang timbul pada saat proses pembuatan kluster adalah karena jumlah tag yang besar (lebih dari 3000 tag), proses klusterisasi berjalan lambat. Untuk mengatasinya, jumlah parameter (dalam hal ini adalah tag) dalam pengelompokan gambar dikurangi menjadi hanya 30 parameter saja. Hal ini tentu saja berdampak pada kurang optimalnya kluster yang terbentuk dan juga pada *association rule* yang di-mining dari tiap kluster.

Tabel 3 menunjukkan tingkat *confidence* dan *support* minimum yang ditetapkan pada setiap kluster. Tabel 3 menunjukkan jumlah *association rule* yang diperoleh untuk setiap kluster. Hanya 4 kluster yang ditampilkan, karena kedua puluh kluster yang lain memiliki hasil yang tidak jauh berbeda.

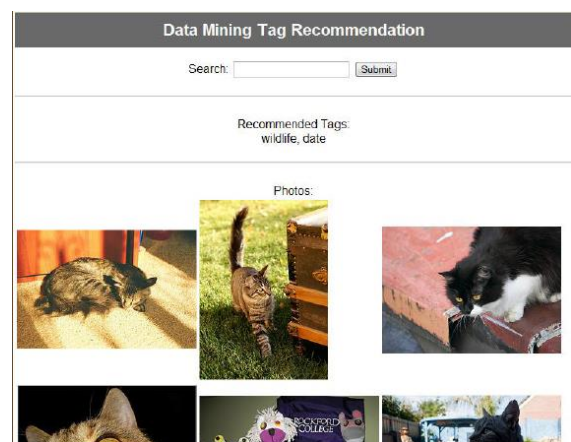
Tabel 3. Tingkat *Confidence* dan *Support* minimum untuk setiap kluster

| Cluster | Min Confidence % | Min Support % |
|------------------------|------------------|---------------|
| Cluster(1): 230 photos | 80 | 12 |
| Cluster(2): 314 photos | 75 | 3 |
| Cluster(3): 229 photos | 75 | 8 |
| Cluster(4): 228 photos | 75 | 17 |

Tiap kluster memiliki nilai *support* minimum yang berbeda karena jika digunakan nilai yang sama akan menghasilkan subset kosong atau bahkan terlalu banyak *frequent itemset*. Nilai *support* yang tinggi dapat menghasilkan subset kosong dan nilai *support* yang rendah menghasilkan terlalu banyak

frequent itemset. Kluster 1 memiliki nilai *confidence* minimum yang berbeda untuk membatasi jumlah *association rule* yang dihasilkan.

Gambar 2 merupakan contoh tampilan aplikasi rekomendasi tag yang telah dibuat. Ketika pengguna mengetikkan kata kunci “cat” di kotak pencarian, maka aplikasi akan melakukan pencarian gambar di database Flickr sesuai dengan kata kunci yang dicari dan menampilkannya dalam bentuk gambar *thumbnail*. Selain itu, hasil rekomendasi tag pun juga akan ditampilkan, dalam hal ini adalah tag “wildlife” dan “date”.



Gambar 2. Contoh Tampilan Aplikasi

III. SIMPULAN

Dari percobaan ini dapat ditarik kesimpulan sebagai berikut:

- Karena besarnya ukuran data, gambar direpresentasikan dalam matriks yang *sparse* sehingga kluster ideal tidak tercapai, dengan kata lain, kluster yang terbentuk tidak seimbang. Penggunaan SVD untuk mengkompresi matriks membantu dalam menyeimbangkan jumlah gambar dalam masing-masing kluster.
- Penggunaan algoritma klasik (Apriori dan K-Means) menyebabkan rendahnya efisiensi proses *mining*.
- Aplikasi untuk menghasilkan rekomendasi tag telah dibuat dan dapat menghasilkan rekomendasi tag yang sesuai dengan kata kunci yang dimasukkan, walaupun masih terbatas pada topik “animal” saja.

Saran untuk kelanjutan penelitian berikutnya adalah:

- a. Menggunakan algoritma *clustering* yang lebih sesuai untuk matriks *sparse*, misalnya Affinity Propagation agar hasil kluster yang terbentuk lebih ideal.
- b. Menggunakan algoritma asosiasi rule yang lain seperti FP-Tree agar hasil asosiasi rule yang terbentuk lebih baik.
- c. Melakukan validasi hasil terhadap urutan rekomendasi tag yang ditemukan dengan menggunakan metrik evaluasi seperti *Mean Reciprocal Rank*, *Success at Rank K*, dan *Precision at Rank K* agar hasil rekomendasi tag yang didapat teruji validitasnya.
- d. Dapat diusulkan untuk mengintegrasikan fungsionalitas rekomendasi tag ini langsung pada situs Flickr®.

IV. DAFTAR PUSTAKA

- [1] M. Ames and M. Naaman. *Why we tag: Motivations for annotation in mobile and online media*. In The conference on Human factors factors in computing systems (CHI'07), pages 971-980, 2007.
- [2] Garg, N., Weber, I.: *Personalized, interactive tag recommendation for flickr*. In: RecSys 2008: Proceedings of the 2008 ACM conference on Recommender systems, pp. 67-74. ACM, New York, 2008
- [3] B. Sigurbjörnsson and R. van Zwol. *Flickr tag recommendation based on collective knowledge*. In The 17th international conference on World Wide Web (WWW'08), pages 327-336, 2008.
- [4] A. Anderson, K. Ranghunathan and A. Vogel. *TagEz: Flickr Tag Recommendation. Association for the Advancement of Artificial Intelligence*. 2008
- [5] Rakesh Agrawal and Ramakrishnan Srikant. *Fast algorithms for mining association rules in large databases*. Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.
- [6] Anonymous, "K-Means Clustering", http://en.wikipedia.org/wiki/K-means_clustering
- [7] Anonymous, "Singular Value Decomposition", http://en.wikipedia.org/wiki/Singular_value_decomposition
- [8] Hicklin J. et al, *JAMA: A Java Matrix Package*, The Mathworks dan National Institute of Standards and Technology (NIST), 2012. [online] <http://math.nist.gov/javanumerics/jama/>
- [9] Wiktionary. JOrtho. 2010